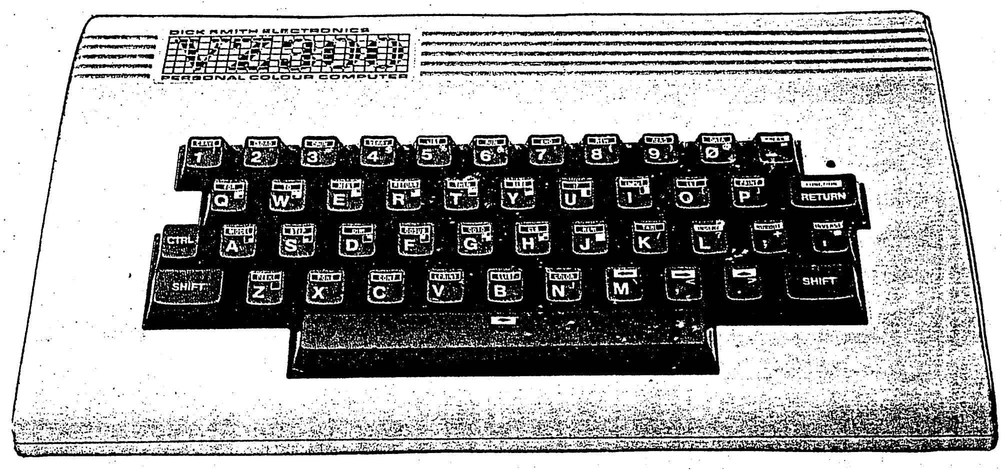
MAY/JUNE 1987 ISSUE # 12

HUNTER VALLEY VZ JOURNAL



=====================================
Y O U R S U B S A R E D U E
=====================================

PRODUCED BY H.V.V.Z.U.G. - -

2 — MY POINT OF VIEW

Welcome to our Anniversary Issue of Hunter Valley VZ Journal. Boy, it's been a hectic first year but we made it thanks to a lot of people who stayed with us through the good times and the bad. I think the regular contributors can be proud of their work in the last twelve months and because of them there are a lot more enlightened VZ users hovering over keyboards all over the country. So Dave Boyce, Robert Quinn, Dave Mitchell, Bob Kitch, Larry Taylor and all those who played a part in the first year the Hunter Valley VZ Users' Group sincerly thanks you and hope you continue the support and high standard you have set.

When we started this Journal we thought of it as a service to our local members, but it seems we hit a winning streak by trying to keep it simple and readable and not too technical. We now have subscribers from all over Australia and New Zealand. Most of the credit for the layout of this Journal belongs to the two Editors we have had during the year. They both have their own styles and ideas and they both put their blood, sweat and tears into the job. Thanks a lot fellers.

Over the past year the Presidents Report has roasted a few people and praised a few and generaly tried to keep Tricky Dicky honest, but alas, I have also put my cloven hoof in my cavernous mouth on the not so seldom occassion and the hot water I got into tended to burn a bit. Having no more fingernails left (for the purpose of slipping burning bamboo slivers) I have decided not to torture our readers any longer so there will be no more reports of this nature from Me (please stop cheering it hurts my ears). Instead you may have to read the Editors comments as his fingernails are intact.

Well now, we've thanked everybody and his brother and now it's time to thank the most important person of all, you, for without your support we could not produce this Journal. We will ask one favor of you though, write to us and let us know what your problems are or what you would like to see in this Mag (Journal, thank you Ed.) as it makes life easier for us if we know what you want. We also dont mind a bit of constructive criticism (put in a nice way of course), so please, give us some feedback and we might have an even better Journal. Now dont forget, your subscribtion is now due for renewal and if you were pleased with the Mag up untill now, in the words of the prophet you aint seen nothin' yet.

You will no doubt notice that the Journal now has printing on both sides of each page, this is because we now have access to a Photocopier and so while the Journal looks thinner it has just as much content if not more and with no loss of quality. This allows us to add more features and pages if we wish to without increasing postage costs.

Recently we had the chance to put on a display at the local Computer Exhibition which we gladly accepted. Considering the short notice (three days) we managed to stage a good show which attracted a fair amount of attention. The increase in VZ sales in the Hunter Valley during the next month or so will be due to our efforts.

Till next time then, bye now.

P.ELLIS (PRES)

Scott Le BRUN - The newsletter is good stuff! The Ram communication addresses are extremely useful. Also Bob Kitch's article (Rom Inv. Char. Set) was very interesting. BUT the article concerning the Magic VZ Eprom is very misleading. The extra Disk "Commands/Functions do not exist in ROM. All this utility does is enable the Basic Interpreter to recognise those particular words. If you try and execute the commands, all you get is an ERROR message. Please correct me if I'm wrong.

Editor (OUCH) - First I would like to thank you for your compliment re the Journal, also I'm sure both Dave and Bob appreciate your comments re their articles. As for the criticism re Magic Eprom I thank you for caring enough in making your comment to what you thought was a very misleading article. As I'm both the Editor and Author of the article hopefully I'll be able to answer to everyone's satisfaction as it is never our intention to be misleading.

I'll quote part of the offending article from issue 8/9, page 11, paragraph 2. "This gives you 19 additional commands/functions and all the disk command exits which are very handy to link any M/L programs".

The key words which I feel you may have missed is "disk command exits". You are quite right in saying the routines do not exist in ROM, exept for SAVE, LOAD, NAME and KILL which are in the Magic Eprom and are used for fast SAVING, LOADING, RENAMING and KILLING of ARRAYS. The rest of the disk command words/exits can be used to link your own M/L rotines. I'll let Robert QUINN'S article on "USING DISK TOKENS" in this issue inform you on how this can be accomplished, thanks Robert. I hope the above has cleared things to everyone's satisfaction.

NOTE :- Since writing the Magic VZ article I've become aware that Steve Olney's and W.Obrist's Extended Basics are not fully compatible. Apparently they use some of the same Tokens for different commands/functions. More on it it in a future issue.

Robert QUINN - I like your Editors Comments page in the Journal. It's well set out for easy reference to the contents. Robert has also supplied some additions to the Ram Communication Addresses which are listed below. Thanks for the compliment and additional information Robert, much appreciated, Ed.

ADDITIONS TO RAM CUMMUNICATION ADDRESSES :-

30780-2	CHARACTER BYTE @ CURSOR POSITION
30889	0 IF CASSETTE INPUT ELSE NON ZERO
30931	LENGHT OF LAST DEFINED STRING
30945	AUTO INPUT FLAG 0=NO AUTO
30943-4	POINTER TO START OF PROGRAM WHEN RUN
30956-7	LINE NO. FOR BASIC LINE ENTERED/DELETED
30965-6	NUMBER OF LINE PROGRAM RUN ENDS ON
30967-8	POINTER TO ADDRESS OF LAST MEMORY CELL SCANNED FOLLOWING BREAK, STOP OR END

NOTE :- We welcome and appreciate your feedback, be it compliment, criticism, hints and tips or requests for help, so keep them coming.

The annual index has been held over till next issue.

As Editor I've made many contacts all over Australia and New Zealand and of course locally and am keeping in touch with many of them. One thing I found that most were more than willing to help out when asked, even when the help requested took a lot of their time and effort. So my special thanks go to Dave Boyce, Dave Mitchell, Robert Quinn, Larry Taylor and Bob Kitch. Of course there were many others who helped out as well. And I'm especially thankful to our Comittee who gave help and support when needed. They showed me what teamwork is all about.

HUNTER VALLEY VZ USERS' GROUP AT NEWCASTLE COMPUTER SHOW :-

Great teamwork and many long tiring hours by the committee enabled us to put on a good show at short notice. Our aim was to attract new members to our club and new subscribers. Also to advertise the fact that there was support for the VZ not just locally, but Nationally as well. How succesfull we were in our efforts we wont know for a while, but we had some feedback already so we could'nt have done too bad a job of it.

Our display consisted of HI-RES slide shows, demonstrations of Dave Mitchell's Extended DOS, our Word Processor Patch and a few other programs and our Journal was for sale at the show too. WE also had on display other VZ publications, LE VZ OOP, VZ USER and VZ DOWN UNDER so prospective subscribers could chose a publication according to their needs.

There was no question about it, the VZ was the cheapest Computer at the show and a lot of people were surprised at it's capabilities. Many people came to the show to select a computer and first time buyers found the VZ very attractive because of the big price difference between the VZ and next cheapest.

One of the most surprising things we learned at the show was the respect other exhibitors at the show held for the VZ even though the computer systems they were demonstrating cost many hundreds if not thousands of dollars.

OTHER VZ USER GROUPS AND PUBLICATIONS :-

VZ USER MARK HARWOOD P.O.BOX 154 DURAL N.S.W. 2158

LE'VZ OOP J.C.E. D'ALTON 39 AGNES St. TOOWONG QLND. 4066

VZ DOWN UNDER C/- GEORGE SEGGIE P.O.BOX 316 ST.KILDA VIC. 3182

VZ-LINK NEWSLETTER P.O.BOX 1972 C.P.O. AUCKLAND NEW ZEALAND

VZ USERS GROUP P.O.BOX 22-094 CHRISTCHURCH NEW ZEALAND

NOTE :- When writing to any above or H.V.VZ. Users' Group for information please enclose a S.S.A.E.

PRESIDENT . . . . PETER ELLIS . . . (049) 69 5697
VICE PRESIDENT . . . . MARK O'BRIEN . . (049) 61 5490
SECRETARY . . . . ROSS WOODS . . . (049) 71 2843
TREASURER . . . . LEIGH ROGERS . . (049) 57 5738
EDITOR . . . . JOE LEON . . . . (049) 51 2756

SUBSCRIPTION - H.V.VZ.JOURNAL - 6 MONTHS $9.00 - 12 MONTHS $18.00

MAIL ALL SUBMISSIONS TO :- HUNTER VALLEY VZ USERS' GROUP
C/- P.O. BOX 161 JESMOND N.S.W. 2299

FOR SALE — QUICKWRITE

* * * NEW * * * NEW * * * NEW * * *

QUICKWRITE WORDPROCESSOR - DISC BASED WORDPROCESSOR A$40.00
QUICKWRITE WORDPROCESSOR IS SUITABLE FOR THE EXPANDED VZ200/300.

QUICKWRITE is software on disc, so RAM and ROM PACKS do not have to be plugged into the VZ which can cause loose port socket connections. QUICKWRITE runs on either the LASER or VZ DOS disc controller. QUICKWRITE saves/loads document text (data) to disc.

FEATURES :-

* Fast disc saving and loading of document text (data).
* Automatic periodic saving of data while in typing mode if required.
* Tape saving and loading of data as a backup medium.
* Loading of E & F tape files (data) possible.
* Printer font changes within the data.
* Capitals/lower case software lock on/off.
* Acommodates wide printers - up to 255 columns.
* A printer/Plotter can also be used.
* Four print justify/wragged modes.
* Adequate operator warnings.
* Labelling of discs allowable, such as date, code, etc.
* The usual editing facilities:-
  Delete, Insert, Find and Replace, Paste, Cut.
* Number 1 or number 2 disc drive selection allowed.

The price of A$40.00 includes surface postage within Australia.
Sold ONLY by VSOFTWAREZ 39 AGNES St., TOOWONG
Q/LAND. 4066. AUSTRALIA

HINT AND TIPS

REJUNIVATING PRINTER RIBBONS :-

Persons using their Dot Matrix Printers a lot find they are forever buying printer ribbons which can become a bit expensive. The life of your printer ribbon can be extended a fair bit. When the printing starts getting a bit faint then it's time to reach for a can of W.D.40 or something similiar.

The technique is simple. First remove printer ribbon from printer and upon examining the case you will notice that the case is made up of of two sections, the case and lid. Using a small kitchen knife prise the lid off being careful not to spill the ribbon on the floor or damage the case as it's fragile. There are two types of ribbons, one has a felt type roller with a small endless loop while the other has a long endless loop with no roller.

Spray the ribbon or roller with W.D.40 being careful not to overdo do it as too much W.D.40 can wash the ink out of the ribbon or roller. Reassemble and your printout should be nice and dark again. W.D.40 may not work with all ribbon types, EG. Carbon, cloth type work OK though. It's possible to reink your ribbon as well using stamp pad ink but I have'nt had much sucess with it so far.

WANTED TO BUY

VZ300 Computer plus DATASSETTE - Phone (049) 69 5697

```
1 '**************************************************
2 '***  DEMO PROGRAM  TO SHOW HOW THE Z80 USES LOGIC ***
3 '***  OPERATIONS TO SET FLAGS - R.B.KITCH 30.11.86 ***
4 '**************************************************
5 :
10 '*** MAINLINE OF PROGRAM ***
20 GOSUB 30000:'*** INITIALIZATION ***
30 GOSUB 20000:'*** PROGRAM MIDDLE ***
40 :
50 B$="[illegible]":X%=3:Y%=15:GOSUB 7000
60 AA$=INKEY$:AA$=INKEY$:IF AA$="" THEN GOTO 60
90 GOTO 30
1000 '***SET NEGATIVE AND ZERO FLAGS***
1100 '***SET NEGATIVE FLAG BY BIT 7***
1200 NG%=ABS(INT(AC%/128))
1300 '***SET ZERO FLAG IF AC%=ZERO***
1400 IF (AC%=0) THEN ZE%=1 ELSE ZE%=0
1990 RETURN
5000 '***DEC N% TO HEX HN$***
5100 HI%=INT(N%/16)
5110 HN$=MID$(HX$,HI%+1,1)
5200 LO%=N%-HI%*16
5210 HN$=HN$+MID$(HX$,LO%+1,1)
5990 RETURN
6000 '***DEC N% TO BIN BN$***
6100 BN$=""
6200 FOR Q%=7 TO 0 STEP -1
6210 BT$="0"
6220 IF(N% AND 2^Q%) THEN BT$="1"
6230 BN$=BN$+BT$
6240 NEXT Q%
6990 RETURN
7000 '***PRINT B$ AT (X%,Y%)***
7100 PS%=32*Y%+X%
7200 PRINT@PS%,B$;
7300 RETURN
8000 '***BIN BN$ TO DEC N%***
8100 N%=0
8200 FOR Q%=0 TO 7
8210 IF VAL(MID$(BN$,Q%+1,1)) THEN N%=N%+2^(7-Q%)
8220 NEXT Q%
8990 RETURN
9000 '***HEX HN$ TO DEC N%
9100 A%=ASC( LEFT$(HN$,1))-48
9110 B%=ASC(RIGHT$(HN$,1))-48
9200 N%=B%+7*(B%>9)-(-1)*(16*(A%+7*(A%>9)))
9990 RETURN
10000 '***AND THE ACCUMULATOR WITH THE DATA***
10100 AC%=AC% AND DT%
10200 GOSUB 1000
10990 RETURN
11000 '***OR THE ACCUMULATOR WITH THE DATA***
11100 AC%=AC% OR DT%
11200 GOSUB 1000
11990 RETURN
12000 '***EXCLUSIVE-OR THE ACCUMULATOR WITH THE DATA***
12100 AC%=(AC% OR DT%) - (AC% AND DT%)
12200 GOSUB 1000
12990 RETURN
```

```
20000 '***PROGRAM MIDDLE***
20100 '***CLEAR SCREEN, PICK CASE, PRINT HEADER***
20110 CLS
20120 MX%=3:MN%=0:GOSUB 40000:CS%=RN%
20130 B$=OP$(CS%):X%=12:Y%=0:GOSUB 7000
20140 B$=HD$:X%=1:Y%=3:GOSUB 7000
20200 '***PRINT TRANSLATIONS OF ACCUMULATOR CONTENTS***
20210 MX%=255:MN%=0:GOSUB 40000:N%=RN%
20250 YO%=5:GOSUB 21000
20260 AC%=N%
20300 '***PRINT TRANSLATION OF DATA ADDRESS CONTENTS***
20310 GOSUB 40000:N%=RN%
20320 YO%=7:GOSUB 21000
20330 DT%=N%
20400 '***GET AND SORT GUESS INTO BIN/HEX/DEC***
20500 B$="":X%=1:Y%=9:GOSUB 7000
20510 INPUT "[illegible] ";G$
20520 IF (LEN(G$)<1) OR (LEN(G$)>9) THEN GOTO 20500
20530 A$=LEFT$(G$,1)
20540 C$=RIGHT$(G$,LEN(G$)-1)
20550 IF A$<>"B" THEN GOTO 20600:'***GUESS NOT BIN***
20560 BN$=C$
20570 GOSUB 8000
20580 F%=3
20590 GOTO 20700
20600 IF A$<>"H" THEN GOTO 20650:'***GUESS NOT HEX***
20610 HN$=C$
20620 GOSUB 9000
20630 F%=2
20640 GOTO 20700
20650 IF A$<>"D" THEN GOTO 20500:'***GUESS NOT DEC***
20655 N%=VAL(C$)
20660 F%=1
20700 '***GUESS HERE***
20710 G%=N%
20800 '***DO OPERATION***
20812 IF CS%=0 THEN GOSUB 10000
20813 IF CS%=1 THEN GOSUB 11000
20814 IF CS%=2 THEN GOSUB 12000
20820 YO%=11:N%=AC%:GOSUB 21000
20900 '***UPDATE SCORE***
20910 IF G%=AC% THEN PT%=PT%+10*F%
20920 B$="[illegible]"+STR$(PT%):X%=9:Y%=13:GOSUB 7000
20990 RETURN
21000 '***TRANSLATE DEC N% AND PRINT AT YO%***
21210 GOSUB 6000
21220 GOSUB 5000
21240 B$=BN$:X%=1:Y%=YO%:GOSUB 7000
21250 B$=HN$:X%=10:Y%=YO%:GOSUB 7000
21260 B$=RIGHT$("  "+STR$(N%),3):X%=14:Y%=YO%:GOSUB 7000
21300 IF YO%<>11 THEN GOTO 21990
21400 B$=STR$(NG%)+STR$(ZE%):X%=17:GOSUB 7000
21990 RETURN
30000 '***INITIALIZATION***
30100 HX$="0123456789ABCDEF"
30120 PT%=0
30130 HD$=" [illegible]"
30200 DIM OP$(2)
30210 FOR Q%=0 TO 2
30220 READ OP$(Q%)
30230 NEXT Q%
30240 DATA "[illegible]","[illegible]","[illegible]"
```

```
30300 GOSUB 50000
30400 CLS
30500 RETURN
40000 '***RANDOM NUMBER GENERATOR IN RANGE OF MN% - MX%***
40100 RN%=INT(RND(0)*(MX%-MN%))+MN%
40200 RETURN
50000 '***INSTRUCTIONS***
50010 CLS
50020 PRINT"THIS PROGRAM IS DESIGNED TO MAKETHE USER FAMILIAR ";
50030 PRINT"WITH LOGIC    OPERATORS."
50040 PRINT"THE 'AND' 'OR' AND 'EOR(XOR)'"
50050 PRINT"OPERATORS ARE TESTED.":PRINT
50060 PRINT"ADDITIONALLY BINARY, HEXADECIMALAND DECIMAL";
50070 PRINT" NOTATION IS USED.":PRINT
50080 PRINT"THE SETTING OF THE 'NEGATIVE'   AND 'ZERO' FLAG ";
50090 PRINT"IN A TYPICAL MPUIS ALSO SHOWN IN THE ANSWER."
50100 PRINT"IF A FLAG IS SET, IT IS SET TO 1";
50110 PRINT"I.E.THE CONDITION IS TRUE.":PRINT:GOSUB 60000
50200 PRINT"THE PROGRAM WILL ACCEPT"
50205 PRINT"RESPONSES PREFIXED BY":PRINT
50210 PRINT"        BINARY  <B>"
50220 PRINT"   HEXADECIMAL <H>  OR"
50230 PRINT"        DECIMAL <D>":PRINT
50240 PRINT"ALSO A BONUS SCORE IS GIVEN IF"
50250 PRINT"THE ANSWER IS GIVEN IN BINARY.":PRINT:PRINT
50255 PRINT:PRINT:PRINT:PRINT
50260 GOSUB 60000:RETURN
60000 '***SUB FOR NEW SCREEN***
60120 PRINT"[illegible]";
60130 AA$=INKEY$:AA$=INKEY$:IF AA$="" THEN GOTO 60130
60140 CLS:RETURN
```

IMPROVING EXTENDED BASIC . . . .

IMPROVING THE EXTENDED BASIC PROGRAM (EBSV2.3) BY D.MITCHELL

The extended program is a very well written program. It can however be improved so it can be used with other extended basics or extended dos. The trouble is it jumps to the DOS interpreter hence no other will work with it but by adding a further six bytes to the program will enable us to do this.

```
BLOAD"EBSV2.3"

7AFA 2A 04 78   LD HL,(7804H)
7AFD 22 3D 7C   LD (7C3DH),HL

31482 42  4 120
31485 34 61 124
```

The easiest way to add the above to the program is by POKING.

```
POKE31482,42:POKE31483,4:POKE31484,120
POKE31485,34:POKE31486,61:POKE31487,124
```

Now all that has to be done is save the program.

```
BSAVE"EBSV2.3",7AFA,8071
```

The modification will work with any program/routine located at 6000-67FF Hex, but not with programs that are top of memory seeking.

SYNTAX ERROR :-

The SYNTAX ERROR is the chief bugbear of beginners and experienced programmers alike. The first error we experienced was more than likely the SYNTAX ERROR. It is the most common error to be found in BASIC programs. A SYNTAX ERROR occurs when the BASIC interpreter either doesn't recognize a "word" or can't make sense of a sequence of characters. (The BASIC interpreter is the program which converts - interprets - your BASIC program into a form directly understood by the computer. It is part of the program contained in ROM.) In other words, a SYNTAX ERROR occurs when you make a typing error, spelling mistake, don't match your parentheses ("()"), or use incorrect punctuation. For example, typing "PRIMT" instead of "PRINT" will cause the dreaded response "SYNTAX ERROR" to leap onto your screen.

However, the "GOOD" thing about SYNTAX ERRORS is that they are the easiest to remedy. When RUNning a program, a SYNTAX ERROR calls a halt to proceedings and outputs its error message. The important part is that we are told in which line the error ocurred. Here's what to do about it:-

LIST the line causing the problem (LIST. will do this for you) and look at it carefully. First, check for typing and spelling mistakes. Often, "1" and "I", "0" and "O" will have been confused. Next, search for unmatched parentheses. Finally, make sure that the punctuation is correct - for help in this respect refer to the VZ Basic Reference Manual, in particular, to pages 161 - 166, or to the examples throughout the manual. If none of this works, find a friend who knows more than you do and ask for help.

Here are a couple of ways to guard against SYNTAX ERRORS creeping into your program in the first place. One of these is to learn how to touch-type, which will be useful to you in more than just getting your programs to run. The reason this will cut down on SYNTAX ERRORS is that you watch the screen rather than the keyboard when typing. For those of us with VZ-200's, the old keyboard should be replaced with a decent one (see VZ Journal issue 5 for information on how to do this). If you can't or won't take this option, then start using the CTRL and CTRL-RETURN key combinations if you don't already (eg use CTRL-Q to get "FOR", use CTRL-RETURN-F to get "RND("). Not only do they limit your typing and spelling ERRORS, but once you learn where to find them, it makes life that much faster.

To finish off, I'd like to make the comment that there's nothing like practice when it comes to spotting errors. Don't forget, there's nothing you can do at the keyboard which will destroy your computer. In the light of that, I present an exercise to try your skill.

EXERCISE :-

Free the following program from SYNTAX ERRORS. One way of doing this exercise and checking if you are right would be to enter it into your computer and keep going until the program runs.

```
10 CLZ
20 MODEO
30 COLOUR 3,0
40 INPUT"GIVE ME TWO NUMBERS,A,B
50 C=(INT(A)*22-B+1I
60 PRIMT A;B;C;C*C
70 MODE 1
80 FOR X=0 TO 20:FPR Y=1 to 30
90 SET (X:Y)
100 NEXT Y,X
110 GOTO 10
120 END
```

NOTE :- The first Person to submit a SYNTAX free version of above program to the EDITOR will have it published in next issue, so get your thinking caps on and get going, (Ed.).

SIDEWAYS BY DAVE BOYCE - - - - -

This Sub-routine READS the HIGH Resolution (Mode 1) SCREEN from the Bottom to the Top and from Left to Right. As it reads the screen, it Prints it to the Dot Matrix Printer. The result is the screen being Printed DOWN the page instead of ACROSS the page. E.G., like turning your screen on it's Right Side.

The Dot Matrix Printer should respond to ESC Code 27 for this Sub-routine to work. If your printer uses different ESC Codes then change the ESC Code 27 in lines 15060 and 15150 to one used by your printer.

In use your program would after the HI-RES screen is drawn send it to the sub-routine via a GOTO or GOSUB to line 15060. After the printout is finished line 15160 should send it back via a GOTO or RETURN to your main program. See back page for an example that this routine can accomplish.

If you have'nt a HI-RES picture then you can use the one from last issue, the ROM INVVERSE CHAR. SET which will do nicely. Add the Sideways Sub-Routine to it with the changes listed below and you'll have your sideways picture.

```
1500 GOSUB 15060
15160 RETURN

15000 '*********************************************
15010 '* HI-RES SIDEWAYS COPY PRINT-OUT SUBROUTINE  *
15020 '* FOR M-1009 AND EPSON COMPATIBLE DOT MATRIX *
15030 '* PRINTERS    -----    BY DAVE BOYCE   -----  *
15040 '*********************************************
15050 :
15060 LPRINTCHR$(27);"A";CHR$(6);
15070 FOR Y%= 0 TO 127
15080 FOR X%=63 TO 0 STEP-1
15090 P=POINT(Y%,X%)
15100 IF P=1 THEN LPRINT" ";:NEXT:GOTO15140
15110 IF P=2 THEN LPRINT"Y";:NEXT
15120 IF P=3 THEN LPRINT"B";:NEXT
15130 IF P=4 THEN LPRINT"R";:NEXT
15140 LPRINT:NEXT
15150 LPRINTCHR$(27);"2"
15160 REM - GOTO XXXX OR RETURN TO MAIN PROGRAM
15170 END
```

12 — ZERO POWER RAM BY JOE LEON

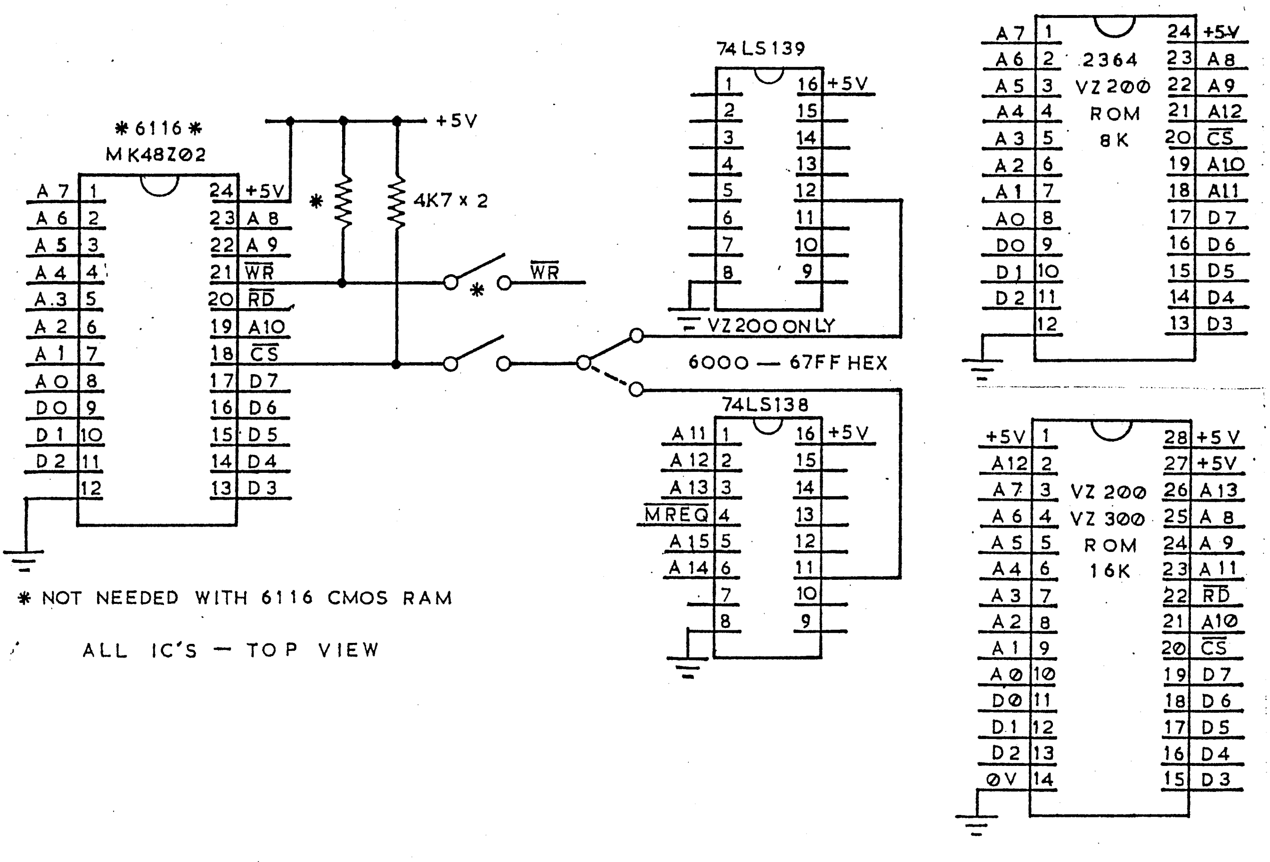
This B.B.RAM project uses the MK48Z02 2K ZERO POWER RAM with built in batteries. It's much simpler to construct than our previous B.B.RAM project. Only 2 Switches and two Resistors are needed. If you have a VZ200 then you can get a CS signal from the internal 74LS139, otherwise a 74LS138 decoder chip is needed. Like the previous project it is designed for the 6000-67FF HEX range. My thanks to Dave Boyce for letting us use the 74LS138 decoding circuit again, thanks Dave.

The easiest way to construct is to piggy back it on an 8K or 16K ROM inside the VZ200/300 or you may prefer to use your own construction methods. The piggyback method of construction is much simpler as most of the connections to the Address and Data buses are automatic when you piggy back. The pinouts of both ROMS are shown. As usual take care as mistakes are easy to make. The reason for the CS switch is that if you have a CART.W.P. then you can switch out the B.B.RAM so the CART.W.P. will work.

I must warn you, the MK48Z02 is expensive at $34.50. You may prefer an ordinary 6116 CMOS 2K RAM at around $4.00 which is pin compatible with the MK48Z02. The choice is yours. If you use a 6116 2K RAM then the WR (WRITE) switch and pull up resistor are not needed and of course you lose any routines stored there when power is switched off, but not with the MK48Z02.

You might ask what use can I put it to. One use is to put Dave Mitchell's Extended DOS in there after first purchasing it of course. See article and ad. elsewhere in this issue. Other utilities that could go there could be Monitors, Extended Basics, Etc. The MK48Z02 is ideal to develop your routines in as you can write and erase as often as you like. When debugged the MK48Z02 can simply be unplugged and put in an Eprom Programmer/Copier for transfer to Eprom.

NOTE :- With the WRITE Switch open the MK48Z02 behaves like an Eprom as you no longer can WRITE to it, only READ. Battery life is estimated at around 3 to 10 years after which it behaves like a normal CMOS RAM.



EXTENDED DOS VERSION 1.0 (C) - COPYRIGHT - DAVE MITCHELL - 1987

COMMANDS :-

MERGE - MERGES basic file from disk with program in memory.

```
DIRA  -  See example - T:MENU      B:PATCH3.1 B:WORDPROC
                       B:EXTDOS E B:EXTDOS R W:DOS-INST
LDIRA - As above, but to screen and printer.

DIRB  - See example  - T:MENU     01 00 7AE9 801B 0532
                       B:PATCH3.1 01 0B 7200 771F 051F
LDIRB - As above, but to screen and printer.

STATUSA  - Prints free disk space to screen on one line.
LSTATUSA - As above, but to screen and printer, see below.
           534 RECORDS FREE   63.500K FREE
```

OLD - Restores a program after using the NEW command.

OLD. - Prints START, END and LENGHT of program in memory in HEX.

DEC XXXXX - Converts DECIMAL to HEX

HEX XXXX - Converts HEX to DECIMAL

STATUSA and LSTATUSA also works with Version 1.0 DOS.

The EXTENDED DOS is available in the two versions below :-

EXTDOS R - T.O.M. SEEKING
EXTDOS E - FOR 2K RAM AT 6000-67FF HEX

Price - $10.00 each or the two for $15.00. Availble from :-

Dave MITCHELL - (079) 27 8519   24 ELPHINSTONE STREET
NORTH ROCKHAMPTON QUEENSLAND 4701

NEWCASTLE AREA :- Joe LEON - (049) 51 2756
22 DRURY STREET WALLSEND N.S.W. 2287

BRIEF USER REPORT by Joe LEON :-

The above utilities work with both V1.0 and V1.2 DOS. At long last V1.0 DOS owners can have a STATUS command. The best thing about EXTDOS R/E is they extend the operating system and if you have a 2K ZERO POWER RAM at 6000-67FF HEX like I have then all above commands are available at power up.

EXTDOS R/E is one of these utilities that once used you don't want to be without. Like any utility it can be improved, certainly not the existing commands, but by adding more. Dave Mitchell sure has done a nice job on it.

DIRA and DIRB are my favourites. The first one displays over 45 files on the screen while the second one gives lots of information on your disk files.

If you read the Zero Power article in this issue it will inform you how to install a 2K CMOS 6116 or ZERO POWER RAM at 6000-67FF HEX so you too can enjoy EXTDOS E.

The program WORDS interprets the WORD TABLE in VZ ROM byte by byte, displaying the entry for each word on a separate line. The address on the left of each entry is the start address of the word in the WORD TABLE, followed by the token code for the word, then the bytes that make up the word. Adjacent to each byte is the character for which that byte is the ASCII code, the first byte of each word being in inverse form to mark the start of a new word. No ASCII interpretation of the zeroes replacing those words that were decommissioned in the VZ is possible or meaningful. Those bytes stand alone.

Press SPACE to freeze the display; press SPACE to continue.

The WORD TABLE still contains locations, in obliterated form, for all the TRS-80 computer's DISK WORDS in the order of their token codes. TRS-80 disk processing routines are not in the VZ ROM. They were not in the TRS-80 ROM either, but were loaded into RAM from disk on startup. In this respect the VZ disk controller with inbuilt ROM DOS is much more convenient.

With this article is a list of all the TRS-80 disk words and relevant data. With the exception of FN, INSTR and TIME$, each of these disk words has an entry in one of two POINTER TABLES. Each pointer consists of a pair of ROM memory cells containing an address in LO HI format (address = LO + HI x 256) and the pointers are in the same sequence in the POINTER TABLES as the corresponding words in the WORD TABLE.

One of the POINTER TABLES begins at address 6178 and contains pointers for all the words with tokens from 128 (END) to 187 (NEW). The other TABLE begins at address 5640 and contains pointers for all the words with tokens from 215 (SGN) to 250 (MID$). Words with tokens from 188 (TAB() to 214 (<) do not have entry to either of the POINTER TABLES.

The address in each pointer is the start of the routine dedicated to processing statements involving the associated word. Since there are no processing routines in VZ ROM for TRS-80 disk words, to what do their pointers point? With the exception of SYSTEM, which points to address 0 (effectively a restart command), and the three words with NO ADDRESS LIST, each disk word points to an address in the COMMUNICATIONS REGION in RAM memory. If you look at the listing, starting from CVI (token 230) and going down to MKD$ (and inserting DEF, token 176, between CVS and CVD) then jumping to CMD (token 133) at the top of the listing and proceeding down, you will see how most of the addresses in the pointers increment by three. There is a gap between KILL and LSET and between RSET and SAVE and between CVI and CVS. Every one of these triples of cells in the communications region (including the triples that fill the gaps just mentioned) contains the same three bytes, which constitute a machine code JUMP instruction (JP 301).

So, if you insert the token for any one of these disk words into a basic program, the basic interpreter will look up the corresponding pointer in the appropriate POINTER TABLE, go to the address in the communications region indicated by the pointer, and execute the JP 301 instruction it finds at that address. Address 301 is the start of a DOS error message routine and so program RUN will terminate with display of the disk error message.

Perhaps the people who designed the VZ200 had it in mind to use these disk words and tokens for VZ disk commands when they got around to marketing a disk drive for the VZ. But, when they eventually designed the VZ DOS, changed their mind and simply set up new tables in the disk controller memory (and so VZ disk words are not tokenised).

Which leaves 23 vacated positions in the WORD TABLE and 23 corresponding pointers in the POINTER TABLES, and more than 23 triples of reserved memory cells in the communications region, all completely unused.

Because all these JP 301 instructions are in RAM, they can be POKE changed to any addresses in memory you please: the first byte of each triple ('195') is the JP instruction; bytes two and three are the address (currently 301) in LO HI format to which the jump is made. If you make up your own machine code routine, POKE the start address of that routine in one of these triples so as to make a JUMP-to-that-address instruction, you can then use the corresponding token in your basic programs (using DLINE) to call and execute your routine. With 23 free tokens, that's a lot of routines you can set up in your VZ.

WORDS

```
10 D=127:FORR=5712TO6175:P=PEEK(R):IFP>128THEND=D+1:PRINT
20 IFP>128THENPRINTUSING"####";R;D;:PRINT":";
30 IFP>128THENPRINTUSING"###";P;ELSEPRINTUSING"##";P;
35 IFP>OTHENPRINTCHR$(P);" ";
40 A$=INKEY$:A$=INKEY$:IFA$<>""THENSOUND20,1:C=NOTC
50 IFC=-1THEN40
60 NEXT
```

WORD ADDRESS	TOKEN	WORD	ADDRESS IN POINTER
5729	133	CMD	31091
5856	162	OPEN	31097
5860	163	FIELD	31100
5865	164	GET	31103
5868	165	PUT	31106
5871	166	CLOSE	31109
5876	167	LOAD	31112
5880	168	MERGE	31115
5885	169	NAME	31118
5889	170	KILL	31121
5893	171	LSET	31127
5897	172	RSET	31130
5901	173	SAVE	31136
5905	174	SYSTEM	0
5917	176	DEF	31067
5976	190	FN	NO ADDRESS LIST
6005	197	INSTR	NO ADDRESS LIST
6015	199	TIME$	NO ADDRESS LIST
6099	230	CVI	31058
6102	231	CVS	31064
6105	232	CVD	31070
6108	233	EOF	31073
6111	234	LOC	31076
6114	235	LOF	31079
6117	236	MKI$	31082
6121	237	MKS$	31085
6125	238	MKD$	31088

16 — MAKING C.W.P. WORK BY JOE LEON

The C.W.Processor will not work with the SUPER II VZ200 (34K RAM) or VZ300 18K Memory Expansion (INTERNAL) as per Technical Report issue 3 of VZ Journal. It will also not work if any RAM (2K) is active at 6000-67FF HEX. VZ owners with the MAGIC EPROM fitted should have no problems as they can switch it out easily if it does'nt work with C.W.P. unless they have extra RAM fitted like I have. So far I have'nt had time to modify my spare VZ with the Magic Eprom. The C.W.P. needs at least 12K of unused Address Range from D000 to FFFE HEX and 2K at 6000 to 67FF HEX to become operational. This article explains how to free the address range needed.

SUPER II VZ200 MODIFICATIONS :-

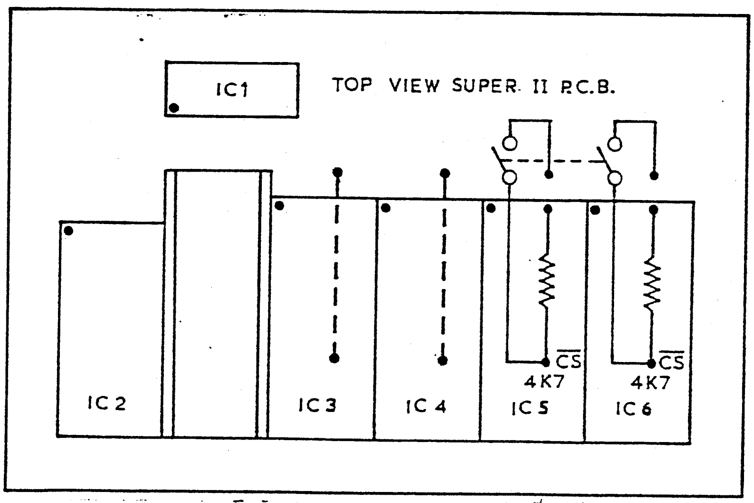
If you study the diagram you will realise the mod. is a simple one, just a couple of pull up resistors and a switch is all that's needed. The CS (Chip Select) line on IC 5 and 6 must be modified as shown. With the switch in the open position both IC 5 and 6 are deactivated allowing the C.W.P. to operate. Make sure the power is off when using the switch.

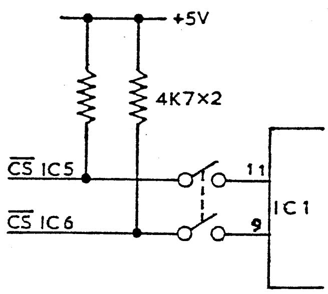
Remove IC 5 and 6 from the board. Next the middle of the sockets have to be cut out so you can remove the two Link Wires. Drill two small holes in the 5V track which is easy to find. It runs under pins 1 and 28 of IC's 3 to 6. Fit the resistors and switches as shown. The switch can be a DPST, DPDT or a Dip Switch. In my case I used a four way Dip Switch which allows me to switch out IC'S 4 to 6 and Zero Power Ram at 6000-67FF HEX. One benefit of being able to switch out the Rams singly is that it allows me to test out VZ200/300 16K Ram Expansion cartridges. The alteration shoud'nt present any problems as it's a simple one.

VZ owners with extra internal RAM get a bonus with the C.W.P. by way of an extra 2K for text after having switched out the top 16K first. EG :-

VZ200 . . . . = 5324 Bytes free for text.
VZ300 . . . . = 15564 Bytes free for text.
VZ200 Super II - 24K = 9420 Bytes free for text.
VZ200 Super II - 16K = 17612 Bytes free for text.

If you have any Ram or Eprom at 6000-67FF HEX it will also have to be deactivated using the same procedure as with Super II. See the Zero Power Ram article in this issue on how to do it.





This new single W.P. Patch supercedes our previous one. It has full compliment of DISK COMMANDS while retaining all TAPE FUNCTIONS. Unlike the earlier version it's childs play to convert your E & F TAPE W.P. It's simply a matter of transfering your TAPE W.P. to the same DISK as the W.P. Patch and BRUNning the Patch which BLOADS your transfered TAPE W.P. To facilitate transfering your E & F TAPE W.P. to DISK a TAPE to DISK transfer program is provided with the W.P. Patch.

As you'll notice from the MENU it will SAVE or LOAD to or from TAPE or DISK allowing you to transfer all your TAPE W.P. Files to DISK or back to TAPE again. It can be used with 1 or 2 DRIVES. Below are the two Menus.

```
VZ-WORD PROCESSOR            VZ-WORD PROCESSOR
=================            =================

  E)DIT TEXT                   L)OAD
  C)LEAR TEXT                  S)AVE
  P)RINT TEXT                  D)IR
  L)OAD FILE                   E)RA
  S)AVE FILE                   R)EN
  V)ERIFY FILE                 I)NIT
  Q)UIT PROGRAM                1-2) DRIVE 1
  D)ISK                        M)ENU

ENTER COMMAND >              ENTER COMMAND >

15042 CHARACTERS FREE        15042 CHARACTERS FREE
```

Dave has done an exellent job with his latest version and the only thing lacking is some fancy printer commands which may be incorporated in a future version.

I've used this and previous W.P. Patches in the compilation of the Journal and the results speak for themselves. When using SAVE, LOAD, REN and ERA upper or lower case letters can be used which makes it easy to back up files using the same file name. Lower case file names are not visible to the normal DIR command but are from within the W.P.

NOTE :- This W.P. Patch is not compatible with the Cartridge Word Processor TAPE or DISK Files or Quickwrite Disk Files. It is possible to transfer Tape Files to Quickwrite though.

SYSTEM REQUIREMENTS :-

DISK DRIVE plus V1.0 or V1.2 Disc Controller
VZ300 plus 16K RAM PACK or
VZ200 plus 18K (16K RAM PACK + 2K) (Minimum)

The price is $10.00 which includes Post/Packing and full instructions and is available from :-

HUNTER VALLEY VZ USERS' GROUP
P.O.BOX 161 JESMOND 2299
N.S.W. AUSTRALIA Phone (049) 51 2756

If you allready have D.SMITH'S E & F Tape Word Processor and recently purchased a Disk Drive then just for $10.00 Dave Mitchell's W.P.Patch will convert it for full DISK use as well and be much cheaper than buying a new Word Processor.

```
100 REM ##########################################################
105 REM # ENHANCED CLS COMMAND BY LARRY TAYLOR 1986              #
110 REM ##########################################################
115 REM # CALCULATE THE NEW TOP OF MEMORY POINTER                #
120 REM ##########################################################
125 NB=79:TM=(PEEK(30897)+PEEK(30898)*256)-NB
130 HB=INT(TM/256):LB=TM-HB*256
135 POKE30897,LB:POKE30898,HB
140 REM ##########################################################
145 REM # RESET THE BASIC STACK POINTER                          #
150 REM ##########################################################
155 CLEAR50
160 REM ##########################################################
165 REM # LOCATION OF SET UP PROGRAM                             #
170 REM ##########################################################
175 EB=31274
180 EH=INT((EB+1)/256):EL=EB+1-EH*256
185 REM ##########################################################
190 REM # LOAD USER EXECUTION PROGRAM POINTER                    #
195 REM ##########################################################
200 POKE30862,EL:POKE30863,EH
205 REM ##########################################################
210 REM # LOAD 22 BYTE SET UP PROGRAM                            #
215 REM ##########################################################
220 FOR T=1TO22
225  READD
230  POKEEB+T,D
235  CS=CS+D
240 NEXT
245 REM ##########################################################
250 REM # GET NEW TOP OF MEMORY AND MOVE TO NEXT LOCATION        #
255 REM ##########################################################
260 TM=PEEK(30897)+PEEK(30898)*256
265 IFTM>32767THENTM=TM-65536
270 REM ##########################################################
275 REM # LOAD 79 BYTE ENHANCED CLS ROUTINE                      #
280 REM ##########################################################
285 FOR T=1TO79
290  READD
295  POKETM+T,D
300  CS=CS+D
305 NEXT
310 REM ##########################################################
315 REM # IF DATA CHECKSUM VERIFIES EXECUTE SET UP PROGRAM       #
320 REM ##########################################################
325 IFCS<>10725THENPRINT"- ERROR IN DATA ENTRY -":END
330 X=USR(0)
335 REM ##########################################################
340 REM # SET UP EXECUTION ROUTINE DATA IN DECIMAL FORM          #
345 REM ##########################################################
350 DATA 49,0,119       :' LD SP,7700H
355 DATA 42,4,120       :' LD HL,(7804H)
360 DATA 34,40,122      :' LD (7A28H),HL
365 DATA 42,177,120     :' LD HL,(78B1H)
370 DATA 35             :' INC HL
375 DATA 34,4,120       :' LD (7804H),HL
380 DATA 205,77,27      :' CALL 1B4DH
385 DATA 195,25,26      :' JP 1A19H
```

```
390 REM ##############################################################
395 REM # ENHANCED CLS ROUTINE DATA IN DECIMAL FORM                   #
400 REM ##############################################################
405 DATA 217            :' EXX
410 DATA 33,91,29       :' LD HL,1D5BH
415 DATA 209            :' POP DE
420 DATA 183            :' OR A
425 DATA 237,82         :' SBC HL,DE
430 DATA 213            :' PUSH DE
435 DATA 217            :' EXX
440 DATA 194,120,29     :' JP NZ,1D78H
445 DATA 229            :' PUSH HL
450 DATA 205,120,29     :' CALL 1D78H
455 DATA 32,7           :' JR NZ,07H
460 DATA 225            :' POP HL
465 DATA 237,91,40,122  :' LD DE,(7A28H)
470 DATA 213            :' PUSH DE
475 DATA 201            :' RET
480 DATA 254,132        :' CP 84H
485 DATA 32,245         :' JR NZ,F5H
490 DATA 35             :' INC HL
495 DATA 126            :' LD A,(HL)
500 DATA 214,48         :' SUB 30H
505 DATA 40,9           :' JR Z,09H
510 DATA 6,8            :' LD B,08H
515 DATA 184            :' CP B
520 DATA 40,4           :' JR Z,04H
525 DATA 16,251         :' DJNZ FAH
530 DATA 24,230         :' JR E6H
535 DATA 209            :' POP DE
540 DATA 209            :' POP DE
545 DATA 17,30,29       :' LD DE,1D1EH
550 DATA 213            :' PUSH DE
555 DATA 35             :' INC HL
560 DATA 229            :' PUSH HL
565 DATA 135            :' ADD A,A
570 DATA 135            :' ADD A,A
575 DATA 135            :' ADD A,A
580 DATA 135            :' ADD A,A
585 DATA 32,1           :' JR NZ,01H
590 DATA 60             :' INC A
595 DATA 198,127        :' ADD A,7FH
600 DATA 33,0,112       :' LD HL,7000H
605 DATA 34,32,120      :' LD (7820H),HL
610 DATA 17,1,112       :' LD DE,7001H
615 DATA 1,255,1        :' LD BC,01FFH
620 DATA 119            :' LD (HL),A
625 DATA 237,176        :' LDIR
630 DATA 225            :' POP HL
635 DATA 201            :' RET
```

AEM 4505 and the VZ 200/300 and Notes to Date (22-3-87). Refer to June/July 1986 issues of Australian Electronics Monthly.

1> Jumpers for USER selectable options
Links U,V,W & X 'should be made'.

2> The I/O Select Link should be made as indicated by the Dotted Line. EG. Pin 11 of IC5 to be joined to the Junction of IC11 pins 8 & 11.

3> All IC's including the 6116 Ram, EXCEPT for IC's 1,2,3 & 9 to be installed.

4> RV1 - the Volume control. Use a Panel mount 10K ohm Lin pot. Dick Smiths R-1806 or R-6806. - Reset sw - D.S.E S-1198. Install Socketts UNDER ALL IC's.

5> Locate the 'PARALLEL PORT' on the top picture on page 88 of AEM June 1986. This picture is the Component Overlay. The P.Port is next to the 'Fingers' which plug into an IBM/Clone. The figure A1 refers only to the Fingers.

The P.Port is numbered 1 to 13. The Front Row of pins (nearest the fingers) carry the Signals and match up with the Centronic/Parallel Port numbers on the VZ Printer Interface Plug.

The GND (earth) can be connected to pin B10 or pin B31 of the IBM fingers (it also appears on pin xx (I cannot tell from the pictures) of the P.Port). You will be able to find which Pin when You buy your P.C.Board, See also fig.3 and Table 2 on page 52 of JULY '86 AEM.

The Printer Interface is used for Interfacing to the Speech Synth. A suitable Power Supply such as the Datasette or Plotter should be used to power the Speech Synth.

NOTES :- This Speech Synthsiser is much easier to use than the one (write-up) which appeared (by myself) in H.V.V.Z.U.G 's Magazine/Journal No#1.

The SPO-256-AL2 chip , LM386 Amplifier chip , the 7805 REGulator and the Interconnecting cable with Centronic's Sockett can be removed from my Speech Synth and used in this (AEM4505) Speech Synthsiser. I also suggest that the Volume control be mounted 'OFF' the Circuit board and ON the front of the Case that you are using.

In the next issue I will show how to incorporate Tone controls as used on my Original Speech Synth. This is relatively easy.

A suit of PUBLIC DOMAIN Programs for the AEM4505 to RUN on the VZ200/300 are available from the H.V.V.Z.U.G or myself.

DAVE BOYCE 41 HEATHER DRIVE CHRISTIE DOWNS 5164 South Aust.

Cover Note - I wrote to AEM some time ago (about 2 months) requesting permission to use 'Parts of' their Diagrams. I haven't heard back from them ?? Considering that if you buy Your CTS chip from tandy's like I did, You will see that the diagrams supplied with their chip are for all Intents and Purposes the same as AEM's.

